ARPANA M RAMASWAMY
DATE: RM18CS164 PAGE:

LAB 08

07/12/2020
Arpana

Q) Implement Dijkstra's Algorithm to find shortest path in graph

```
mat[20][20];
void dijkstra(int src, int dest)
{
  // store distance and route
  int dist[n], visited[n], route[n];

  for(int i=0; i<n; i++)
  {
    dist[i] = 9999;
    visited[i] = 0;
    route[0] = -1;
  }

  dist[src] = 0;
  for(i=0; i<n-1; i++)
  {
    int u = minimum(visited, dist);
    visited[u] = 1;
    for(int v=0; v<n; v++)
    {
      if(!visited[v] && mat[u][v] != 9999 &&
```

ARPANA M RAMASWAMY
1RM18CS164
DATE: 07/06/2020 PAGE:
(Arpana)

```
dist[u] != 9999 && (dist[u] + mat[u][v] < dist[v])
{
    dist[v] = dist[u] + mat[u][v];
    route[v] = u;
}
}
}

cout << "shortest distance is " << dist[dest];
cout << "shortest path is ";
path (route, src);
}

// To print path
void path (int route[], int i)
{
    if (route[v] == -1)
        return;
    path (route, route[i]);
    cout << i << " ";
}
```